REMARKS

The Examiner is thanked for his/her careful and very thorough Office Action. The Examiner is particularly thanked for the helpful suggestions regarding correction of the alleged informalities.

Claims 14, 15, 17, and 20 are allowed. Claims 1-5, 7-13, 16, and 19 have been rejected. Claim 18 has been objected to. By the foregoing amendments, various Claims are sought to be amended or canceled without prejudice.

Note that the amendments to Claims 2 and 7 are intended to be purely formal amendments, and are believed not to change the scope of these claims. Specifically, Claim 2 was an inadvertent switch of two terms and the corrected version of the claim uses the same language that is found in the specification.

Claim 11 has been canceled.

Claims 21 and 22 are added claims that are supported by applicant's original disclosure. Specifically, Claims 21 and 22 are supported by the original specification, page 8, lines 15-30. No new matter is added by this amendment.

Review of the References

Some of the major technical differences between the references applied and the disclosure of the present application will now be reviewed. Of course, these points in the specification do not define the scope or interpretation of any of the claims; they are listed merely to help appreciate the importance of the claim distinctions that will be reviewed thereafter.

Mizuyabu

Mizuyabu et al (U.S. Patent No. 6,297,832), hereafter referred to as "Mizuyabu", relates to a way of sequencing memory accesses so that an "optimal ordering pattern typically includes sequencing alternating accesses between the two banks of the memory such that when a page fault is occurring in one bank of memory, a memory access is occurring in the opposing bank." (Abstract) Mizuyabu does not disclose how the access is to take place, only a way of sequencing the accesses.

Peddada

Peddada et al. (U.S. Patent No. 6,295,068), hereafter referred to as "Peddada", relates to a 3D graphics driver for a host that manages a texture cache in the local graphics memory on the host side. The driver disclosed by Peddada does not appear to manage texture memory in the main memory nor does it suggest managing page faulting of texture data. Peddada also does not disclose allowing the graphics accelerator itself direct access to the main memory or any other innovations with regard to graphics accelerators.

Slaughter

Slaughter et al (U.S. Patent No. 6,202,146), hereafter referred to as "Slaughter", relates to software for endianness checking for platform-independent device drivers. Slaughter is not a graphics processor component, nor does it reference or appear to manage page faults in memory.

Duluk

Duluk, Jr. et al (U.S. Patent No. 6,288,730), hereafter referred to as "Duluk", refers to a deferred graphics pipeline processor comprising a texture unit and a texture memory associated with the texture unit. The texture unit applies texture maps stored in the texture memory, to pixel fragments. The textures are MIP-mapped and comprise a series of texture maps at different levels of detail, each map representing the appearance of the texture at a given distance from an eye point. Duluk does not reference or appear to manage page faults in memor.

If the undersigned attorney has overlooked a relevant teaching in any of the references, the Examiner is requested to point out very specifically where such teaching may be found.

Rejections under 35 U.S.C § 112

I. Claim 2 satisfies the requirements of §112.

The present amendments make this rejection moot. The Examiner rejected Claims 2 and 8-11 under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The Examiner has suggested that the claim contains subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant are that the inventors, at the time of the application was filed, had possession of the claimed invention.

The Examiner rejected Claim 2 under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. For the purpose of clarity, claim 2 as amended is reproduced below:

> 2. A computer system, comprising:
> a graphics accelerator unit which manages page faulting of texture data, from main memory used by at least one host processor into a dedicated graphics memory, invisibly to the host processor, except when said graphics accelerator unit calls for data which has not recently been present in said main memory.

To satisfy the written description requirement, the claimed invention need not be expressed ipsis verbis in the original specification. In re Wertheim, 541 F.2d 257, 262, 190 U.S.P.Q. 90, 96 (C.C.P.A. 1976) ("It is not necessary that the application describe the claim limitations exactly, . . . but only so clearly that persons of ordinary skill in the art will recognize from the disclosure that appellants invented processes including those limitations."); In re Wright, 866

F.2d 422, 425, 9 U.S.P.Q.2d 1649, 1651 (Fed. Cir. 1989) ("[T]he claimed subject matter need not be expressed in haec verba in the specification in order for that specification to satisfy the written description requirement."). The present amendment clarifies the claim and includes the language to show that the graphics accelerator unit manages page faulting of texture data, from **main memory used by at least one host processor into a dedicated graphics memory.** Therefore, the presently amended claim 2 complies with written description requirement.

The Examiner rejected dependent Claims 8-10, which depend directly from independent Claim 2 and incorporate all the limitations thereof, under U.S.C 112, first paragraph, as failing to comply with the written description requirement. Therefore, these rejections have been overcome.

Rejection Under 35 USC 103(a)

Claims 1, 3 and 13 stand rejected under 35 USC Section 103(a) as being unpatentable over Mizuyabu in view of Slaughter et al.

II. The combined references do not teach or suggest "a graphics accelerator unit which manages page faulting of texture data invisibly to the host processor," as claimed.

The asserted combination of references does not teach or suggest each limitation of Claim 1. Specifically, Claim 1 recites:

> **1. A computer system, comprising: a graphics accelerator unit which manages page faulting of texture data invisibly to the host processor.**

Mizuyabu relates to sequencing memory accesses in a video graphics system in order to minimize the effects of page faults. Mizuyabu does not appear to disclose any innovations with regard to graphics accelerators that relieve the host processor of the duties associated with duty of dealing with those page faults, i.e. managing page faults "invisibly to the host processor."

Applicant respectfully suggests that examiner has assigned different definitions of "manages page faulting," and "invisibly" to Mizuyabu than are used in the present context.

First, Mizuyabu only provide a "method and apparatus for sequencing memory accesses'" (col. 2 lines 15-16) to minimize time delays caused by page faults and not a way of managing the page faults by fetching the data in order to free the host processor to execute other tasks.

Second, Mizuyabu does not claim to invisibly do anything, rather Mizuyabu claim only to minimize time delays so that "page faults are effectively hidden" (col. 2, line 17). In this context, "hidden" means to reduce or eliminate a timing delay, and does not appear to relieve a host processor of any tasks or burdens during page faults, as is taught in the present application.

In addition, Mizuyabu does not "manage" the actual page faults, but simply sequence them in a bifurcated memory to minimize time delays. Mizuyabu does not appear to offload any functions from the host processor so as to free the main CPU up to perform other functions. This is perhaps due to the fact that Mizuyabu are concerned only with minimizing time delays caused by page faults and not with offloading work from the main CPU:

> ...memory controller 20 ***receives*** requests for memory access from the plurality of clients and organizes these requests in such a way as to minimize page faults... (col. 5,11. 1-3) [emphasis added]

The Mizuyabu patent is directed towards the memory controller receiving requests and teaches away from the present invention that manages page faults. Specifically, Mizuyabu "receives," whatever data created the page fault. The present invention recognizes the problem of managing page faults. Mizuyabu is directed towards minimizing the effects of the faults rather than managing how data is retrieved when a page fault occurs. By contrast, the present inventions are concerned with managing page faults, and specifically, the importance of offloading work from the main CPU by performing the page managing operations for the host processor via the graphics accelerator in some embodiments. Therefore, one of ordinary skill in the art would not be motivated to combine or modify the references in the manner required to form the solution disclosed in the claimed invention. Accordingly, the present innovations solve problems that are not even considered by the video graphics system of Mizuyabu.

Further, Mizuyabu relates to a way of sequencing memory accesses so that an "optimal ordering pattern typically includes sequencing alternating accesses between the two banks of the memory such that when a page fault is occurring in one bank of memory, a memory access is occurring in the opposing bank" (Abstract). Because Mizuyabu does not address how to retrieve data that was not found, Applicants must assume that conventional methods are employed by Mizuyabu to retrieve data when a page fault occurs. Absent some teaching or suggestion in the cited reference that something else happens, examiner may not read applicants invention into the disclosure of Mizuyabu. To do so is impermissible hindsight.

In contrast, the presently claimed inventions does not employ known or traditional methods to retrieve data when a page fault occurs, but instead teaches innovative ways to retrieve and manage, as claimed.

Moreover, Mizuyabu does not disclose how the access is to take place, only a way of sequencing the accesses. Mizuyabu uses the phrase "page fault penalty" to describe the preparation time needed to prepare the memory to access a new page in a single bank of memory.

> Although the page fault timing penalty is incurred by the tiled memory access, while this timing penalty is being incurred the data for the linear client is being actively read from the second bank. (col. 6, lines 25-28)

By contrast, the present invention teaches innovative systems and methods for relieving the host processor of the processing burden caused by a page fault.

> When a texture page fault occurs the Texture Read Unit interfaces with a Texture DMA Controller to actually get the data. (p. 65, lines 19-20)

Therefore, the Mizuyabu definition of "page fault penalty" is a time delay, whereas the present application describes a page fault as having the host retrieve a page of texture from the second level of memory, i.e. the host's physical memory and does not necessarily fetch the data any faster. This is because Mizuyabu does not relieve the host processor of any tasks normally performed when a page fault occurs, such as:

> a. Determine where the page is located in host physical memory.
> b. Determine which page out of the working set (in level 1memory) to use.
> c. Make this page the most recently used page (as well as continuing to keep the least-recently-used list up to date as other pages are used).
> d. Update the page tables for the new page and remove any reference to the page just bumped out of memory (if any).

e. Download the page.
f. Restart texture processing.
(p. 8, lines 17-18, original application)

In the present case, the Examiner has not cited teaching in the reference that discloses the limitations of Claim 1. Therefore, Applicant respectfully submits that Claim 1 is not taught or suggested by Mizuyabu in view of Slaughter. Therefore, the rejection is respectfully traversed because all limitations of the claimed invention must be considered when determining patentability. In re Lowry, 32 F.3d 1579, 1582, 32 U.S.P.Q.2d 1031, 1034 (Fed. Cir. 1994).

Claim 3 also recites features not supported by the asserted combination. Specifically, Claim 3 recites:

> 3. A computer system, comprising: at least one CPU, operatively connected to have read/write access to a main memory; first memory management logic, which virtualizes said main memory with reference to at least one bulk storage unit; and a graphics accelerator unit, comprising rendering accelerator logic, dedicated graphics memory, and a second memory management unit which manages texture data for said accelerator logic and performs page faulting of said texture data, invisibly to said CPU.

Again, Mizuyabu do not appear to disclose a graphics accelerator unit that "performs page faulting of said texture date, invisibly to said CPU." As stated earlier, the operations of memory controller could not be determined to be invisible to the CPU nor be said to manage page faults. Rather, Mizuyabu teaches towards sequencing page faults in an attempt to minimize time delays, not towards managing page faulting. In addition, claim 3 recites "a second memory management unit which manages texture data for said accelerator logic." This innovative second memory management unit does not appear to be contained

within any of the cited prior art. Mizuyabu teaches away from using a second memory management unit because it is only capable of sequencing the page faults, not managing the memory. In contrast, claim 3 teaches a separate memory management unit that "performs page faulting of texture data."

Therefore, for the reasons stated above, the Examiner has not cited teaching in the reference that discloses the limitations of Claim 3. Accordingly, Applicant respectfully submits that Claim 3 is also not anticipated by Mizuyabu in view of Slaughter.

Finally, claims 13 and 12 depend directly from independent Claim 3 and incorporates all the limitations thereof, also include additional limitations that are not shown or suggested by the asserted combination.

III. The combined references do not teach or suggest a graphics accelerator unit which manages page faulting of texture data invisibly to the host processor.

Claims 2, and 8-10 stand rejected under 35 U.S.C. Section 103(a) as being unpatentable over Mizuyabu and Slaughter in view of Peddada. The asserted combination of references does not teach or suggest each limitation of Claim 2. Specifically, Claim 2 recites:

> 2. A computer system, comprising: a graphics accelerator unit which manages page faulting of texture data, from dedicated graphics memory into a main memory used by at least one host processor, invisibly to the host processor, except when said graphics accelerator unit calls for data which has not recently been present in said main memory.

Applicant reurges the arguments and reasons stated above, specifically

directing the examiners attention to those arguments regarding the management of page fault data done invisibly to the host processor. Applicant respectfully submits that Claim 2 is also not anticipated by Mizuyabu and Slaughter et al. in view of Peddada.

Thus, for the reasons discussed above, Applicant respectfully requests withdrawal of this rejection.

Finally, dependent Claims 8-10, which depend directly from independent Claim 2 and incorporate all the limitations thereof, also include additional limitations that are not shown or suggested by the asserted combination.

Therefore, for the reasons stated above, Applicant respectfully submits that Claims 8-10 are also not anticipated by Mizuyabu and Slaughter et al. in view of Peddada Thus, for the reasons discussed above, Applicant respectfully requests withdrawal of these rejections.

IV. The combined references do not teach or suggest a "graphics memory manager that fetches texture data automatically," as claimed in claim 4 and claim 7.

Claims 4, 5, 7, 16 and 19 stand rejected under 35 U.S.C. Section 103(a) as being unpatentable over Peddada (U.S. Patent 6,295,068) in view of Duluk, Jr. et al. (U.S. Patent 6,288,730).

The asserted combination of references does not support each limitation of Claim 4. Specifically, Claim 4 recites:

> 4. A computer system comprising: a host processor having respective physical memory associated therewith; and a graphics accelerator unit having respective local memory associated therewith, and also having a graphics memory

> manager; wherein, when said graphics accelerator unit attempts to access texture data which is in said physical memory associated with said host, said graphics memory manager fetches said texture data automatically.

Peddada relates to a 3D graphics driver that manages a texture cache in the local graphics memory. The driver disclosed by Peddada does not suggest managing page faulting of texture data. Peddada also does not disclose allowing the graphics accelerator itself direct access to the main memory or any other innovations with regards to graphics accelerators. Peddada specifically say that:

> "The list of textures includes pointers to the texture in the AGP portion of the main memory and pointers to the texture in the texture cache. The 3D graphics engine is unable to read textures from the main memory. The 3d graphics engine is only able to read textures from the texture cache." (col. 4, lines 29-31)

Examiner cites Duluk et al. as disclosing an invention that uses a processor to render 3D graphics. Duluk et al. mention that "the texture memory has texture data stored and accessed in a manner which reduces access conflicts " (abstract), but does not teach anything relating to managing page faults. These references do not teach any innovations, neither combined not independently, in the field of graphics accelerators with respect to managing page faults

Some of the present applications innovations disclose a computer system in which, when a logical page fault occurs and the page of texture is in the host's physical memory it will be fetched automatically by the graphics memory manager, and the host is not aware anything has happened. One of the novelties of the present inventions is that the graphics accelerator unit actually fetches the data and puts it in the cache so that the host processor never knew that a page fault would have occurred had the data not been fetched by the graphics accelerator.

All of this is done automatically, i.e. without prompting from the host processor, and invisibly, i.e. the host processor is unaware that the data was in a position that would have caused a page fault.

Applicant respectfully submits that Claim 4 is also not anticipated by Peddada in view of Duluk. Thus, for the reasons discussed above, Applicant respectfully requests withdrawal of this rejection.

Dependent Claim 5, which depends directly from independent Claim 4 and incorporate all the limitations thereof, also include additional limitations that are not shown or suggested by the asserted combination.

Claim 7 also recites features not supported by the asserted combination. Specifically, Claim 7 recites:

> 7. A computer system comprising: a host processor having host physical memory associated therewith, and also having virtual memory management; and a graphics accelerator unit having respective physical memory associated therewith, and also having virtual memory management; and wherein, when said graphics accelerator unit attempts to access texture data which is in said host physical memory, if said texture data is in said host physical memory, said graphics accelerator unit fetches said texture data there from automatically; and if said texture data is not in said host physical memory, said texture data is first loaded into said host physical memory, and thereafter said graphics accelerator unit fetches said texture data automatically from said host physical memory.

Again, neither independently nor in combination, do the cited references teach towards the present innovations. The present innovations teach a graphics processor that automatically fetches data without using the conventional processing capabilities of the host processor.

Therefore, for the reasons stated above, Applicant respectfully submits that

Claim 7 is also not anticipated by Peddada in view of Duluk et al. Thus, for the reasons discussed above, Applicant respectfully requests withdrawal of this rejection.

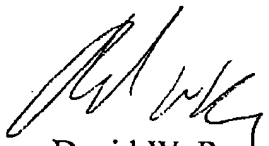
Dependent Claim 16, which depends directly from independent Claim 4 and incorporate all the limitations thereof, also include additional limitations that are not shown or suggested by the asserted combination. Therefore, for the reasons stated above, Applicant respectfully submits that Claim 16 is also not taught or suggested by Peddada in view of Duluk et al. Thus, for the reasons discussed above, Applicant respectfully requests withdrawal of this rejection.

Dependent Claim 19, which depends directly from independent Claim 7 and incorporates all the limitations thereof, also include additional limitations that are not shown or suggested by the asserted combination.

Conclusion

This amendment is being submitted in response to the non-Final Office Action dated 6/21/2005 and, therefore, could not have been submitted earlier. Its entry is respectfully requested. All grounds of rejection and/or objection are traversed or accommodated, and favorable reconsideration and allowance are respectfully requested. Thus, all grounds of rejection and/or objection are traversed or accommodated, and favorable reconsideration and allowance are respectfully requested. The Examiner is requested to telephone the undersigned attorney or Robert Groover for an interview to resolve any remaining issues.

Respectfully submitted,



David W. Roe,
Reg.No. 55,785
Customer Number 29106
Attorney for Applicant

Groover & Holmes
One Galleria Tower, Suite 1370
13355 Noel Road
Dallas TX 75240
972-980-5840, fax -5841

20 September 2005